УДК: 004.8: [519.1+629.7] https://doi.org/10.15407/jai2024.04.242

О. В. Савчук[1], О. М. Моргаль[2] , А. Ю. Мулярчук[3]
[1,2,3] Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Україна
пр. Берестейський, 37, м. Київ, 03056
[1]savchuk_l1@ukr.net
[2]m_olegm@ukr.net
[3]mularcukanastasia@gmail.com
[1]https://orcid.org/0000-0003-3176-7952
[2]https://orcid.org/0000-0003-4654-9499

ВИКОРИСТАННЯ ЛОГІЧНИХ МЕТОДІВ ПРИ ДІАГНОСТУВАННІ БАГАТОАГЕНТНОЇ СИСТЕМИ КОНДИЦІОНУВАННЯ

O. Savchuk[1], O. Morgal[2], A. Muliarchuk[3]
[1,2,3] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine
37, Beresteyskyi Ave., Kyiv, 03056
[1]savchuk_l1@ukr.net
[2]m_olegm@ukr.net
[3]mularcukanastasia@gmail.com
[1]https://orcid.org/0000-0003-3176-7952
[2]https://orcid.org/0000-0003-4654-9499

LOGICAL METHODS USAGE IN THE MULTI-AGENTS AIR-CONDITIONING SYSTEM DIAGNOSTICS

Анотація. Розглянута ідея про можливість застосування кооперативних агентів для дослідження та діагностування бортової системи кондиціонування. Запропонована логічна модель пошуку несправностей I та II типу. Дана робота дозволяє надати можливість оволодіти практичними знаннями та навичками побудови логічних моделей першого та другого типів для отримання мінімального тесту працездатності та пошуку місця несправностей і пошкоджень на етапах розробки, експлуатації та технічного обслуговування складних інформаційних систем. Розроблений алгоритм, що поєднує розробку узагальненої логічної моделі I типу з боку входів функціональної схеми системи та подальшу побудову логічної моделі II типу з її вільних виходів. Метод глибокого навчання дозволяє підвищити точність моделі, швидкодію знаходження несправностей, передбачення та запобігання аварій.

Ключові слова: бортові системи кондиціонування, логічні методи, агенти, діагностика.

Abstract. The possibility of cooperative agents usage for the on-board air conditioning system research and diagnosis is considered. A logical model for type I and II faults searching is proposed. This work provides an opportunity to master the practical knowledge and skills the first and second types logical models building to obtaining a minimum test of performance and finding malfunctions and damages the place at complex information systems development, operation and maintenance the stages. An algorithm has been developed that combines the development a generalized I type logical model from the system functional circuit input side and the subsequent construction II type a logical model from its free outputs. The deep learning metod allows to increase the accuracy of the model, the speed of finding faults, predicting and preventing accidents.

Keywords: on-board air conditioning systems, logical methods, agents, diagnostics.

Вступ

Безвідмовне та життєстійке працювання систем кондиціювання (СК), зокрема бортових СК, є актуальною задачею, особливо за часів воєнного часу. Ці системи забезпечують комфорт у приміщеннях, підтримуючи оптимальну температуру, тиск та рівень вологості.

Для підтримки справного стану та вірного функціонування бортових СК необхідно мати певне діагностичне забезпечення, що на всіх етапах його життєвого циклу повинно включати: номенклатуру діагностичних параметрів і/або ознак; методи і засоби діагностування; правила та алгоритми діагностування.

Кожному методу та етапу діагностування, звичайно, відповідає своя діагностична модель СК. Рішення про вид технічного стану може прийматися

методами логічного аналізу, ідентифікації чи розпізнавання образів згідно з вимогами, умовами та інформаційними можливостями діагностичних моделей [1].

При застосуванні логічних методів діагностування на етапі отримання первинної діагностичної інформації про динамічні системи рішення про їх стан доцільно приймати методами логічного аналізу. При цьому можна використовувати *логічну* діагностичну модель, що побудована на базі функціональної схеми системи залежно від причинно-наслідкових зв’язків блоків між собою та виду використаної інформації.

Ефективність використання логічних моделей при діагностуванні динамічних об’єктів пов’язана з достатньою повнотою відображення в них взаємозалежності і належності значень вхідних і вихідних параметрів областям їх допустимих робочих значень.

Вибраний підхід полягає у тому, що після визначення верхніх і нижніх граничних допусків значень на параметри чи ознаки здійснюється перехід до двійкової логічної моделі. Це дозволяє формалізувати і значно поліпшити вибір діагностичних параметрів і побудову відповідних алгоритмів діагностування. Визначені допуски можуть збігатися з межами справності, працездатності чи вірного функціонування або мати відносно них деяке упередження.

Щоб побудувати логічну модель, необхідно поділити систему на складові частини, кількість яких обирається залежно від заданої глибини пошуку пошкоджень та адекватності вибраної діагностичної моделі поставленій меті діагностування.

У загальному випадку такому поділу підлягає багато електричних та добре структуризованих систем, функціональні схеми яких відомі чи можуть бути розроблені. Для таких систем складається логічна модель I типу з відображенням усіх існуючих зв’язків між функціонально визначеними складовими частинами.

В іншому випадку логічна модель може бути задана орієнтованим графом, вершини якого відповідають обраним складовим частинам системи, а також її входам і виходам, а дуги — зв’язку між складовими частинами та зовнішніми входами і виходами. Кожній дузі відповідає окремий вхідний, внутрішній або вихідний параметр системи згідно з функціональними зв’язками між її складовими частинами.

Але існують системи, які не мають чіткої структуризації, тоді виникає необхідність використання графів причинно-наслідкових зв’язків між параметрами і ознаками, що визначать її технічний стан. Саме граф причинно-наслідкових зв’язків є формалізованою *логічною моделлю* II типу.

У побудові логічної моделі на основі функціональної схеми системи використовують поняття істотного входу. Істотність входів фізично полягає в тому, що вихід справної (працездатної) складової частини може виявитись за межами припустимих значень, якщо залишиться хоч один із істотних для даного виходу входів.

Другим важливим поняттям при побудові логічної моделі динамічних СК є узгоджена система допусків. Це означає, що за умови послідовного з′єднання блоків, допуски на вихідні параметри попереднього блока повинні збігатися з допусками на вхідні параметри наступного блока або упереджувати їх. При паралельному з′єднанні блоків допуски усіх суттєвих входів також мають бути узгодженими.

Мета *аналізу логічної моделі* полягає у виборі діагностичних параметрів чи ознак і в побудові алгоритмів діагностування. Аналіз логічної моделі передбачає побудову таблиці функцій пошкоджень за певними правилами. При цьому виявляються групи блоків, які охоплено зворотним зв’язком. Оскільки за умов діагностування не можна без розриву зворотних зв’язків відрізнити, який блок в такій групі пошкоджено, то кожен з них об’єднується в узагальнений блок. Початкова логічна модель перебудовується в узагальнену.

Далі розглядаються правила вибору діагностичних параметрів і/або ознак: 1) для перевірки справності, працездатності,

вірності функціонування системи; 2) для пошуку поодиноких (одноразових) дефектів або пошкоджень, яким вважається дефект (пошкодження) одного з блоків логічної моделі.

Завершальний етап - наводяться алгоритми пошуку пошкоджень [1, 5, 6]. Алгоритми перевірки справності, працездатності, вірності функціонування не складні. Справді, досить упевнитися, що діагностичні параметри чи ознаки, що використовуються для перевірки, перебувають у межах допусків, щоб перевірку технічного стану вважати завершеною.

Алгоритм пошуку поодиноких пошкоджень дозволяє знайти пошкоджений блок логічної моделі, а потім і пошкоджену складову частину системи. Алгоритм пошуку пошкоджень довільної кратності полягає в послідовному виявленні пошкоджених блоків логічної моделі з наступною заміною відповідних їм складових частин СК на справні доти, доки не виявлятимуться і не будуть усунені всі пошкодження. Багатофункціональними вважаються такі вироби, алгоритм функціонування яких складається з кількох підалгоритмів.

Аналіз логічної моделі, незалежно від форми її зображення, проводиться за єдиними правилами: 1) побудова таблиці функцій пошкоджень (несправностей); 2) вибір діагностичних параметрів для перевірки справності, працездатності, вірності функціонування; 3) вибір діагностичних параметрів або ознак для пошуку поодиноких пошкоджень або відмов.

Правила вибору діагностичних параметрів чи ознак забезпечують *мінімізацію* їх обліку. Методика встановлює алгоритми пошуку поодиноких пошкоджень та пошкодження довільної кратності. Для реалізації алгоритмів діагностування на системі потрібно забезпечити можливість контролю діагностичних параметрів. Передбачається, що сама система має бути контролездатною згідно держстандартів.

Методика вибору діагностичних параметрів або ознак стосовно систем, що подаються логічними моделями, поширюється на одно- та багатофункціональні системи дискретної або неперервної дії різної фізичної природи, щодо яких можна застосовувати допускові методи контролю.

Огляд багатоагентних систем

В [2] поданий огляд багатоагентних систем з прикладами задач, які вони дозволяють вирішувати: по-перше, це рішення проблеми розпізнавання стану навколишнього середовища, зокрема, завдання розміщування штучних агентів розпізнавання, по-друге, поняття та виконання інтелектуального агента – для розв’язання управлінських завдань [3]. Практичний інтерес для СК мають штучні агенти функціонального призначення.

Штучні агенти можуть базуватися на логічному програмуванні, нейронних мережах, які є основною складовою глибокого навчання, а також виконувати кооперативне розв’язання задач [4]. Роботи [5, 6] присвячені принципово важливим розв’язанням завдань системного аналізу багатофакторних ризиків: функціонування системи замкнутого оборотного водопостачання, розподілу напруги бортової мережі, декомпозиції систем в послідовність системно узгоджених задач з метою підвищення їх живучості.

Метою даної статті являється побудова логічних моделей багатоагентної бортової системи кондиціювання для отримання мінімальних діагностичних тестів працездатності, розробки алгоритму пошуку несправностей, апробування метода глибокого навчання для покращення діагностування.

Правила побудови логічної моделі об’єкта діагностування

Систе́ма кондиціюва́ння пові́тря (СКП) - одна з бортових систем життєзабезпечення. СКП призначена для підтримання тиску і температури повітря в кабіні літального апарата на рівні, що забезпечує нормальну життєдяльність пасажирів. Герметичність кабін забезпечується їх конструктивним виконанням, наявністю ущільнень на

дверях і люках, а також постійним наддувом від СКП [7-10].

В аварійних випадках, для швидкого припинення наддуву кабіни, наприклад, під час пожежі двигуна і надходження диму з повітропроводів у кабіну, в СКП передбачають аварійні заслінки, які практично миттєво перекривають трубопроводи, або триходові крани, які в нормальному режимі плавно керують заслінкою на відкриття-закриття, а в аварійному режимі електродвигун працює у форсованому режимі тільки на закриття [8].

Розглянемо випадок завдання СКП блочною функціональною схемою (БФС) при реалізації логічної моделі I типу.

Позначимо:

P_i — i-й блок функціональної схеми, в якому i= $\overline{1,L}$ — номер блока; N — загальна кількість блоків;

X_{ij} — j-й зовнішній вхід i-го блока, в якому j=$\overline{1,K}$ — номер зовнішнього входу;

К — кількість зовнішніх входів;

Y_{il} — 1-й внутрішній вхід i-го блока, що є виходом другого блока, в якому l = $\overline{1,L}$ — номер внутрішнього входу;

L — кількість внутрішніх входів;

Z_{im} — m-й вхід i-го блока, де m = $\overline{1,m}$ — кількість виходів.

Зауваження. У разі необхідності скорочення обсягу логічної моделі групи блоків мають бути об'єднані у більші блоки.

Функціональна схема перебудовується в перетворену функціональну схему з розгалуженням. Для цього треба керуватись правилом: якщо вхідний (вихідний) сигнал блока функціональної схеми охарактеризований кількома фізичними параметрами, то кожен з них треба позначити окремим входом (виходом) блока.

Наприклад, якщо блок P_1 має вхідні сигнали X_i та Y_i і вихідний сигнал Z_i, кожен з яких охарактеризований двома фізичними параметрами (амплітудою, фазою чи частотою), то після перебудови цей блок матиме чотири входи X_{i1}, X_{i2}, Y_{i1}, Y_{i2}, і два виходи — Z_{i1} та Z_{i2}. Певно, якщо деякий вихід Z_i блока P_i, що є входом Y_k блока $P_к$, розгалужується на декілька виходів Z_{ij}, то вхід Y_k теж розгалужується на таку саму кількість входів Y_{kj}.

Будується перетворена функціональна схема СКП, в якій є розщеплені входи і виходи усіх блоків P_i, i = $\overline{1,N}$, що з'єднуються між собою згідно з первинною функціональною схемою. Задаються для всіх входів і виходів блоків P_i можливі області значень фізичних параметрів, що допускаються чи не допускаються. Для кожного виходу блока P_i виділяються істотні входи, тобто ті, які впливають на значення виходів, що допускаються чи не допускаються.

Кожен блок P_i з m ≥ 2 виходами перетвореної функціональної схеми поділяється на m блоків, що мають по одному виходу та істотні входи. Блоки утвореної схеми позначаються символами Q_i, i = $\overline{1,n}$, де n — кількість блоків логічної моделі.

Блоки логічної моделі з'єднаються між собою і з зовнішніми входами СКП зв'язками згідно з початковою функціональною схемою. Позначаються зовнішні входи і виходи моделі.

Потім треба перевірити:

а) чи збігаються визначені області значень параметрів виходу Z_i та входу Y_i, які допускаються або не допускаються при послідовному з'єднанні блоків Q_i та Q_j;

б) чи збігаються області допустимих значень вхідних параметрів Y_i та Y_j (X_i і X_j), якщо блоки Q_i та Q_j паралельно приєднуються до виходу блока Q_i або до зовнішнього входу моделі.

У логічній моделі позначення внутрішніх входів Y_i заміняються на позначення зв'язаних з ними виходів Z_i.

Оскільки СКП багатофункціональний, то логічна модель будується для кожного його підалгоритму окремо. За цих умов щоразу використовується відповідна частина функціональної схеми, що бере участь у роботі. Ознаки позначених параметрів входів і виходів тих самих складових частин, що беруть участь у реалізації різних підалгоритмів функціонування, кожного разу можуть бути особливими і відрізнятися від ознак інших підалгоритмів.

Після побудови моделі 1-го типу для її використання потрібно виконати певні умови:

а) для всіх елементів схеми СКП перелічити всі можливі пошкодження та визначити спочатку поодинокі, а потім кратні пошкодження;

б) для усіх розглянутих пошкоджень перелічити всі види технічного стану, в яких може перебувати СКП, тобто від множини пошкоджень перейти до множини станів;

в) для об'єкта перелічити всі можливі елементарні перевірки із вказівками місця перевірок;

г) домовитися про необхідні і достатні зовнішні впливи / дії на виконання можливих елементарних перевірок у робочих або в тестових режимах.

Якщо всі перелічені умови виконані, то логічна модель 1-го типу може / повинна мати такі якості:

а) якщо в СКП відсутні зворотні зв'язки, то всякі його пошкодження можна спостерігати та розрізняти. Якщо ж неможливо уникнути показу зворотних зв'язків, то не всі пошкодження спостерігатимуться та розрізнятимуться;

б) для логічних моделей без зворотних зв'язків принципово існує можливість підібрати певну послідовність елементарних перевірок, яка дозволяє виділити будь-яку комбінацію кратних пошкоджень;

в) якщо в СКП спостерігаються тільки поодинокі пошкодження, то його логічну модель можна задати деревоподібним спрямованим графом, на основі якого побудувати всі або вибіркові тестові набори елементарних перевірок для виявлення будь-якого поодинокого пошкодження;

г) якщо у логічній моделі СКП не можна уникнути показу зворотного зв'язку, то діагностичний набір елементарних перевірок може бути побудований тільки тоді, коли при діагностичному експерименті можна розірвати його на будь-який контактний вплив.

Якщо зазначені умови не виконуються, то частина функціональної схеми, яка охоплена зворотним зв'язком, виявляється недіагностованою і не може бути перевіреною відносно стану кожного блока, що зв'язаний зворотним зв'язком.

Правила побудови таблиці станів, або функцій пошкоджень

На основі логічної моделі будь-якого типу для випадку поодиноких пошкоджень по всіх блоках схеми (або вершинах) скласти *таблицю функцій станів* або таблицю функцій пошкоджень, в яких рядки утворюють всю множину технічних станів S за поодинокими пошкодженнями

$$S = \{ S_i \},\ S_i \in S,\ i = \overline{0,n}, \qquad (1)$$

де S_i — i-й технічний стан при пошкодженні в одному блоці (або елементі) схеми; S_0 — справний стан, коли немає жодного пошкодження СКП.

Стовпці таблиці функцій станів показують множину елементарних перевірок

$$\Pi = \{ \Pi_j \},\ \Pi_j \subset \Pi,\ j = \overline{1,n}, \qquad (2)$$

де j — j-та елементарна перевірка, яка визначає реакцію j-го блока при пошкодженні в цьому блоці чи в попередніх блоках.

На перетині i-го рядка з j-м стовпцем визначається результат A_{ij} елементарної перевірки Π_j при стані S_i, яка може бути допустимою (логічна одиниця) або недопустимою (логічний нуль). Отже, таблиця функцій станів може подаватися як матриця результатів перевірок у вигляді множини

$$A = \{ A_{ij} \},\ A_{ij} \subset A,\ i = \overline{0,n},\ j = \overline{1,n}. \qquad (3)$$

Множина A описує СКП у просторі елементарних перевірок. Сама таблиця є зручною моделлю для побудови діагностичних наборів перевірок. Використання методу покриття за сумою по mod2 дозволяє визначити необхідний і достатній набір перевірок, який визначає стан системи або вказує місце пошкодження з точністю до елементарного блока.

1) Побудувати рядки таблиці функцій пошкоджень і замінити кожну групу

однакових рядків одним рядком, який називається узагальненим. Позначити останній символами об'єднаних рядків. Перевірки досить позначити номером одного з об'єднаних рядків.

2) За тим самим правилом об'єднати однакові стовпці таблиці. Отримана після об'єднання рядків і стовпців таблиця називається *узагальненою* таблицею функцій пошкоджень.

Зауваження. Фізично існування груп однакових рядків означає, що пошкодження блоків, які становлять одну групу, не відрізняються між собою.

Отже, маємо вихідну логічну модель.

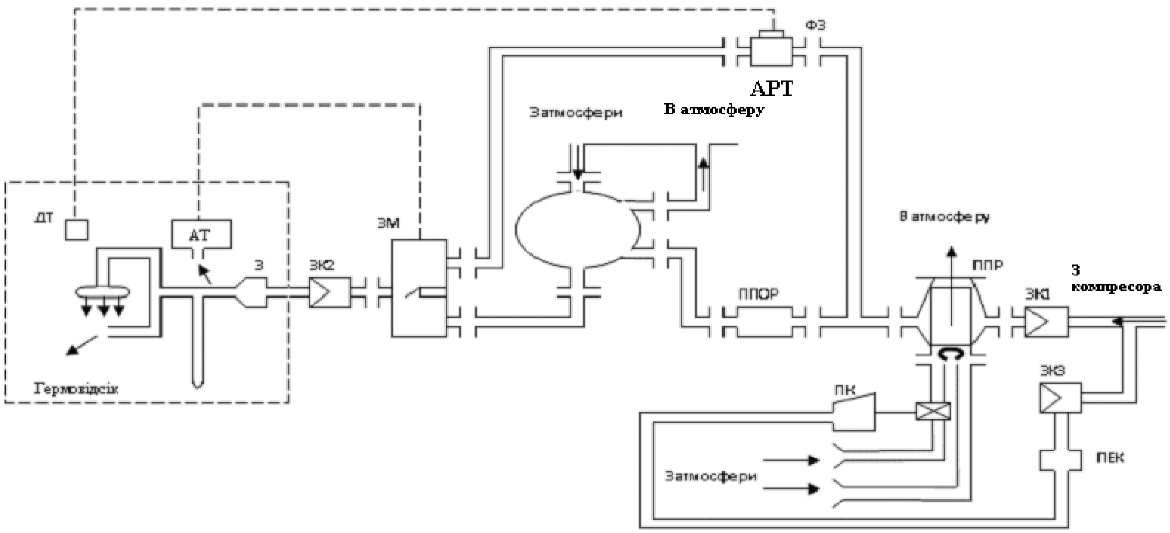


Рис. 1. Принципова схема кондиціонування повітря бортового гермовідсіку

Ця система (рис. 1) є однофункціональною. Система кондиціонування повітря автоматично підтримує тиск і температуру в герметизованому відсіку. Для наддуву, охолодження та нагрівання герметизованого відсіку використовується стиснуте гаряче повітря, відібране від компресора двигуна. Це повітря через зворотний клапан ЗК1 (штучний агент 1) надходить у повітряно-повітряний радіатор ППР, де може охолоджуватися повітрям, зібраним з атмосфери. На лінії продування ЛП ППР є зворотний клапан ЗК3 (штучний агент 3) і пневмоелектричний клапан ПЕК (агент), що керується ззовні та вмикає додаткову продувку ППР атмосферним повітрям. У разі потреби зниження температури у гермовідсіку ПЕК керує гермоклапаном ПК (агент). З ППР повітря потрапляє в магістральний повітропровід, який на кінці розгалужується на "холодну" та "гарячу" лінії через фланцеві з'єднання ФЗ.

В "холодній" лінії повітря охолоджується повітряно-повітряним охолоджуючим радіатором ППОР і турбохолодильником ТХ. Далі повітря через змішувач ЗМ і зворотний клапан ЗК2 (штучний агент 2) проходить у гермовідсік.

У "гарячій" лінії встановлено автоматичний регулятор тиску АРТ, який підтримує тиск повітря в гермовідсіку залежно від сигналу датчика тиску ДТ.

Повітря, що проходить "холодною" та "гарячої" лініями, потрапляє до змішувача повітря ЗМ, який є додатковим органом автомата регулювання температури АТ в гермовідсіку. За сигналом автомата АТ ЗМ відчиняє заслінку З "холодної" або "гарячої" лінії залежно від температури в охолоджуваному (обігрівальному) гермовідсіку. Із змішувача ЗМ повітря через зворотний клапан ЗК2 (штучний агент 2) потрапляє до заслінки З вимикання наддува, а потім у колектори гермовідсіку.

Вважатимемо перелічені агрегати системи кондиціонування блоками функціональної схеми (рис. 2) і позначимо їх так:

P1 = ЗК1;	P5 = ЗМ;	P9 = ПЕК;
P2 = ППР;	P6 = ЗК2;	P10 = ПК;
P3 = ППОР;	P7 = З;	P11 = АРТ;
P4 = ТХ;	P8 = ЗКЗ;	P12 = АТ.

Зовнішніми входами є такі:

X1, X8 (X1 = X8) – тиск повітря, що відбирається від компресора;
X2 – температура повітря, що відбирається від компресора;
X9 – сигнал керування клапаном ПЕК;
X10 – теплоємність атмосферного повітря на вході в ПК. Оскільки температура повітря не впливає на роботу ЗК1, ЗКЗ та ПЕК, то вона не розглядається як вхідні сигнали блоків P1, P2, P9.

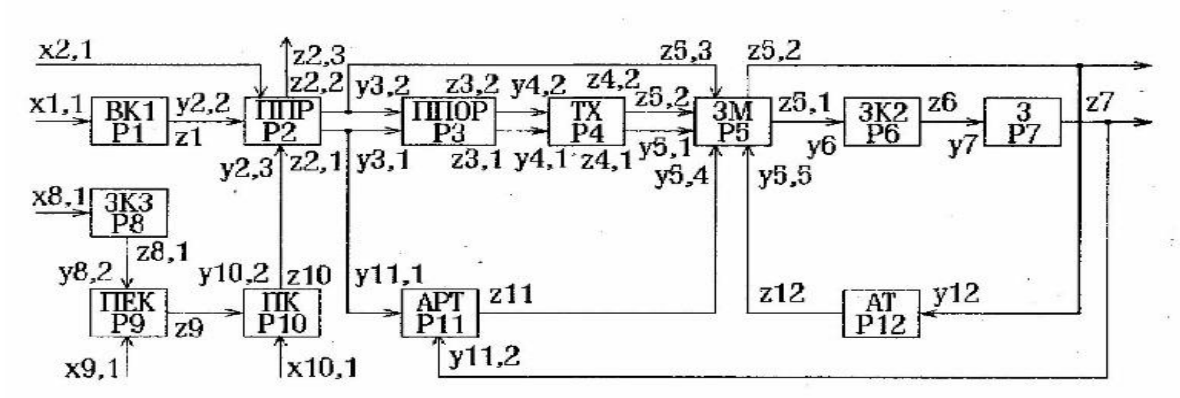


Рис. 2. Функціональна схема системи кондиціонування повітря

Блоки мають входи (X_j, Y_k) та виходи Z_i:

P1	X1	– тиск вхідного повітря, що відбирається від компресора;
	Z1	– тиск вихідного повітря;
P2	X2	– температура вхідного повітря;
	Y2,2	– теплоємність атмосферного повітря, що продуває ППР при відчиненні ПК;
	Z2,1	– тиск вихідного повітря;
	Z2,2	– температура вихідного повітря;
	Z2,3	– теплоємність повітря, що випускається в атмосферу після продування ППР;
P3	Y3,1	– тиск вхідного повітря;
	Y3,2	– температура вхідного повітря;
	Z3,1	– тиск вихідного повітря;
	Z3,2	– температура вихідного повітря;
P4	Y4,1	– тиск вхідного повітря;
	Y4,2	– температура вхідного повітря;
	Z4,1	– тиск вихідного повітря;
	Z4,2	– температура вихідного повітря;
P5	Y5,1	– тиск холодного повітря;
	Y5,2	– температура холодного повітря;
	Y5,3	– температура вхідного гарячого повітря (з ППР);
	Y5,4	– тиск гарячого повітря;
	Y5,5	– величина переміщення заслінки З змішувача ЗМ за сигналом від АТ;
P6	Y6	– тиск вхідного повітря;
	Z6	– тиск вихідного повітря;
P7	Y7	– тиск вхідного повітря;
	Z7	– тиск вихідного повітря;
P8	Y8	– тиск вхідного повітря;
	Z8	– тиск вихідного повітря;
P9	Y9	– тиск вхідного повітря;
	X9	– сигнал керування клапаном;
	Z9	– тиск вихідного повітря;

P10:
- X10 – теплотримання атмосферного повітря на вході ПК;
- Y10 – тиск вхідного повітря, що відчиняє ПК;
- Z10 – теплотримання атмосферного повітря, що продуває ППР при відімкненні ПК;

P11:
- Y11,1 – тиск вхідного повітря;
- Y11,2 – тиск повітря в гермовідсіку;
- Z11 – тиск вихідного повітря;

P12:
- Y12 – температура повітря у гермовідсіку;
- Z12 – величина переміщення заслінки змішувача 3 за сигналом від АТ.

Зробимо заміну блоків перетвореної функціональної схеми СКП блоками логічної моделі (рис. 3):

P1→ Q1;	P5→ Q9, Q10;	P9 →Q14;
P2→ Q2,Q3,Q4;	P6→ Q11;	P10→Q15;
P3→ Q5,Q6;	P7→ Q12;	P11→ Q16;
P4→ Q7, Q8;	P8→ Q13;	P12→ Q17.

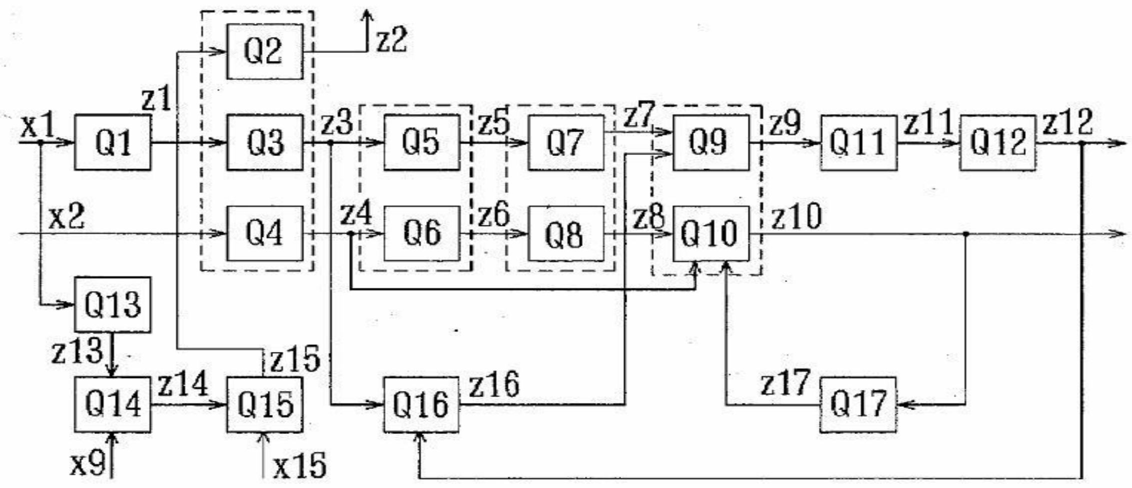


Рис. 3. Логічна модель системи кондиціонування повітря

Таблиця 1. Функції станів системи кондиціонування

Пі Si	П1	П2	П3	П4	П5	П6	П7	П8	П9	П10	П11	П12	П13	П14	П15	П16	П17
S0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S1	0	1	0	1	0	1	0	1	0	1	0	0	1	1	1	0	1
S2	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S3	1	1	0	1	0	1	0	1	0	1	0	0	1	1	1	0	1
S4	1	1	1	0	1	0	1	0	1	0	1	1	1	1	1	1	0
S5	1	1	1	1	0	1	0	1	0	1	0	0	1	1	1	0	1
S6	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	0
S7	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	0	1
S8	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0
S9	1	1	1	1	1	1	1	1	0	1	0	0	1	1	1	0	1
S10	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0
S11	1	1	1	1	1	1	1	1	0	1	0	0	1	1	1	0	1
S12	1	1	1	1	1	1	1	1	0	1	0	0	1	1	1	0	1
S13	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1
S14	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1

Пі Si	П1	П2	П3	П4	П5	П6	П7	П8	П9	П10	П11	П12	П13	П14	П15	П16	П17
S15	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
S16	1	1	1	1	1	1	1	1	0	1	0	0	1	1	1	0	1
S17	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0

Побудуємо таблицю функцій пошкоджень для СПК, яку задано логічною моделлю (рис. 3). Для цього підготуємо квадратну матрицю з кількістю рядків та стовпців, що дорівнює кількості блоків логічної моделі, тобто 17 x 17, і додамо ще один рядок S_0 для справного стану. Рядки цієї матриці позначимо символами станів S_i ($i = \overline{0,17}$) блоків логічної моделі, а стовпці – символами перевірок $П_j$ ($j = \overline{1,17}$) за виходами цих блоків.

На перетині і-го рядка та j-го стовпця виявляється результат a_{ij} елементарної перевірки $П_j$ за станом S_i, що може бути допустимим (логічна одиниця) або недопустимим (логічний нуль). Отже табл.1 є матрицею результатів у вигляді множини:

$$A = \{a_{ij}\},\ a_{ij} \in A,\ i = \overline{1,17},\ j = \overline{1,17}, \quad (4)$$

де S_0 – це справний стан, що утворюється з одиниць. Далі припустимо, що перший блок логічної моделі є пошкодженим, тоді в табл. 1 $a_{11} = 0$. Тепер виявляємо всі значення перевірок $П_i$ ($i = \overline{1,17}$), що перевіряють виходи блоків, які пов'язані з першим блоком, і за даними умовами дорівнюють нулю. Інші результати перевірок позначимо одиницями. Таким чином заповнюємо перший рядок табл. 1. Щоб заповнити другий та наступні рядки таблиці, для і - го рядка треба послідовно задавати результати перевірок $a_{ij} = 0$ ($i = j$) та позначити результати перевірок, що пов'язані з відповідними блоками, нулями. Клітинки таблиці, що залишаються, слід позначити одиницями. Коли таблицю станів заповнено, будуємо першу таблицю покриття (табл. 2) у вигляді квадратної матриці 17x17, де результати перевірок порівняно з табл.1 інвертовані. Порівняємо попарно рядки таблиці покриття і відшукаємо рядки, що збігаються. В табл. 2 збігаються рядки 9, 11, 12, 16 та 10, 17. Побудуємо *узагальнену таблицю покриття* (табл. 3), об'єднавши рядки і стовпці, що збігаються. Узагальнений рядок S09, 11, 12, 16 розміщуємо замість рядка S09. А рядок S010, 17 – замість рядка S010.

Таблиця 2. Перша таблиця покриття системи кондиціонування

Пі Si	П1	П2	П3	П4	П5	П6	П7	П8	П9	П10	П11	П12	П13	П14	П15	П16	П17
S01	1	0	1	0	1	0	1	0	1	0	1	1	0	0	0	1	0
S02	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S03	0	0	1	0	1	0	1	0	1	0	1	1	0	0	0	1	0
S04	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	1
S05	0	0	0	0	1	0	1	0	1	0	1	1	0	0	0	1	0
S06	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	1
S07	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	1	0
S08	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1
S09*	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1	0
S010^	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
S011*	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1	0
S012*	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1	0
S013	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0

Пі Si	П1	П2	П3	П4	П5	П6	П7	П8	П9	П10	П11	П12	П13	П14	П15	П16	П17
S014	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
S015	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
S016*	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1	0
S017^	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1

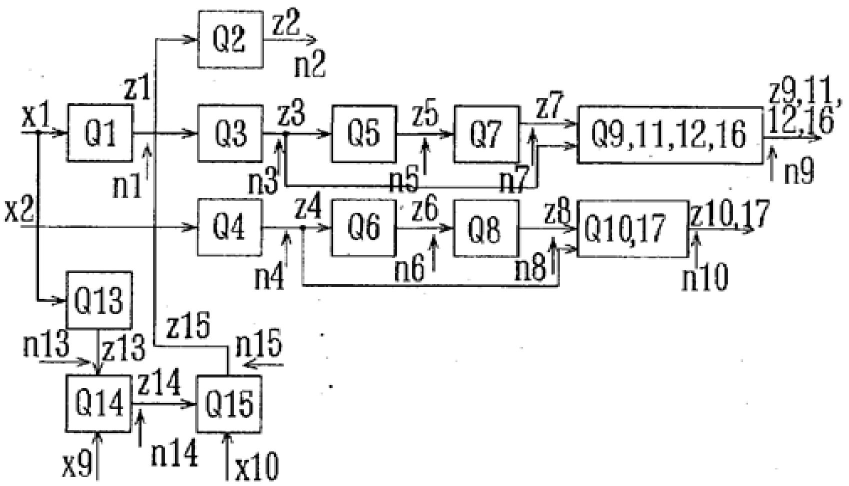


Рис. 4. Узагальнена логічна модель системи кондиціонування

Таблиця 3. Узагальнена таблиця покриття системи кондиціонування

Пі S0i	П1*	П2*	П3	П4	П5*	П6*	П7*	П8*	П9*, 11, 12, 16	П10* (17)	П13*	П14*	П15*
S01	1	0	1	0	1	0	1	0	1	0	0	0	0
S02	0	1	0	0	0	0	0	0	0	0	0	0	0
S03	0	0	1	0	1	0	1	0	1	0	0	0	0
S04	0	0	0	1	0	1	0	1	0	1	0	0	0
S05	0	0	0	0	1	0	1	0	1	0	0	0	0
S06	0	0	0	0	0	1	0	1	0	1	0	0	0
S07	0	0	0	0	0	0	1	0	1	0	0	0	0
S08	0	0	0	0	0	0	0	1	0	1	0	0	0
S09, 11, 12, 16	0	0	0	0	0	0	0	0	1	0	0	0	0
S010, 17	0	0	0	0	0	0	0	0	0	1	0	0	0
S013	0	1	0	0	0	0	0	0	0	0	1	1	1
S014	0	1	0	0	0	0	0	0	0	0	0	1	1
S015	0	1	0	0	0	0	0	0	0	0	0	0	1

Правила вибору діагностичних параметрів тестів логічної моделі І типу

Будуємо першу таблицю покриттів, що виходить у такий спосіб: $S_0 \oplus S_i = S_{0i}$, де $\oplus$ - сума за модулем 2 [1].

Перша таблиця покриття будується на основі порівняння результатів перевірок між справним станом S_0 і кожним пошкодженим станом $S_1, \ldots, S_n$. Отриманий результат записується на перетині рядка і стовпця на основі порівняння результатів перевірок за модулем 2. Якщо результати різні (1 та 0), маємо логічну одиницю, а якщо однакові – логічний нуль. Побудову діагностичного набору перевірок починають з пошуку рядків, в яких розміщується по одному позитивному результату. Набір перевірок є достатнім,

якщо їх результати, що позначені логічними одиницями, перекривають усі n рядків. Мінімальний набір перевірок, що перекривають усі n рядків, є елементарним *діагностичним тестом працездатності*.

Для СПК за табл. 3 отриманий такий мінімальний діагностичний тест працездатності $П_2$ U ($П_9$, $П_{11}$, $П_{12}$, $П_{16}$) U ($П_{10}$, $П_{17}$).

Зауваження. Згідно з правилами побудови першої таблиці покриття, у клітинках головної діагоналі таблиці завжди будуть одиниці. Крім того, серед рядків таблиці завжди знайдеться такий, у якому існує єдина одиниця. Йому відповідають блоки узагальненої моделі, виходи яких є так званими вільними виходами, тобто зовнішніми виходами, які не можуть бути входами ніяких блоків.

Для пошуку місця несправності будується друга таблиця покриття $S_i \oplus S_j = S_{ij}$, де i,j = $\overline{1,n}$ [1]. Для цього треба порівняти попарно усі рядки узагальненої таблиці покриття на множині стовпців, які відповідають номерам всіх виходів.

Зауваження. Кількість порівнянь в узагальненій таблиці з n рядками дорівнює числу комбінацій

$$C_n^2 = \frac{n(n-1)}{2}. \qquad (5)$$

Є сенс порівняти ці пари рядків на множині всіх стовпців узагальненої таблиці покриття, тобто для табл. 3. Для кожної пари рядків, що порівнюються, вибрати у вигляді логічної суми символи перевірок (номери стовпців таблиці), на яких рядки пари відрізняються між собою.

Приклад. Якщо рядок S_{03} відрізняється від рядка S_{06} на перевірках (стовпцях) $П_5$, $П_8$, $П_{10}$, то логічна сума записується як $П_5$ U $П_8$ U $П_{10}$. Це означає, що для розрізнення рядків S_{03} та S_{06} досить контролювати вихід Z_5, або Z_8, або Z_{10}.

Об’єднати логічні суми всіх пар рядків у логічний добуток і, використовуючи правила алгебри логіки, перетворити до вигляду *логічної суми логічних добутків*. Знайти в утвореному виразі найменший за кількістю співмножників добуток.

Доповнити список діагностичних параметрів такими, що відповідають виходам, із символів яких складено логічний добуток.

Для табл. 3 друга таблиця покриття має 78 строк (тут не наводиться), і знайти сукупність перевірок, що перекривають всі строки другої таблиці покриття треба програмно. Може бути декілька тестів для пошуку поодиноких несправностей. Вибираємо *мінімальну* сукупність елементарних перевірок, що контролюються для пошуку поодиноких пошкоджень.

Остання кодова таблиця має строками перелік блоків системи, а кількість стовпців – число перевірок мінімального тесту пошуку несправностей. Розпізнавання стану системи виконується порівнянням значень перевірок отриманого кода з відповідною строкою кодової таблиці [1].

Для багатофункціональної системи необхідно виконати алгоритм для кожного режиму функціонування, далі отримані діагностичні параметри об’єднати в загальний список.

Логічна модель II типу

Система зображується у вигляді графа причинно-наслідкових зв’язків. Тоді в ній виділяються вхідні параметри X_j, j = $\overline{1,L}$, внутрішні та зовнішні параметри Z_i, i = $\overline{1,n}$ (або ознаки, події, явища), які характеризують технічний стан СКП.

Граф G1 будується з входів до виходів. Треба зіставити одиницю між параметрами X_j та n параметрами Z_i (L + n) вершин графа. Розглянути всі впорядковані пари параметрів (X_j, Z_i) та (Z_i, Z_s), j = $\overline{1,L}$; s, i = $\overline{1,n}$, $s \neq i$, та якщо вихід із області допустимих значень першого параметра пари безпосередньо (без зміни значень інших параметрів) тягне за собою вихід із області допустимих значень другого параметра пари, то вершини, що відповідають цим параметрам, з’єднують дугою, напрямленою від вершини першого параметра до вершини другого. Позначити вершини графа, які є параметрами Z_i, символами Q_i. Перевірити виконання умов у підрозділах стосовно пар

параметрів, яким відповідають вершини, що з'єднуються у графі дугою (ребром).

Для багатофункціональних систем треба окремо побудувати графи для кожного підалгоритму функціонування.

Використати вихідну логічну модель для побудови узагальненої логічної моделі 2-го типу за правилами:

а) кожну групу вершин графа, якій відповідає узагальнений рядок в узагальненій таблиці функцій пошкоджень, замінити у логічній моделі на одну узагальнену вершину графа;

б) входами узагальненої вершини є всі входи відносно групи об'єднаних вершин;

в) узагальнена вершина має один вихід, що можливо розгалужується, і його дуги є зовнішніми відносно групи об'єднаних вершин виходами.

Узагальнена вершина відрізняється індексами об'єднаних вершин. Виходи Z_i, Z_j, Z_k, подані як узагальнений вхід $Z_{i,j,k}$, і мають дві гілки, які відповідають зовнішнім виходам Z_s та Z_p. Згідно з цим можна зробити дві перевірки — $П_s$ та $П_p$.

Алгоритм побудови графа причинно-наслідкових зв'язків

Розглянемо граф G2, що будується з виходів до входів. Для даної системи, згідно рис. 3, маємо три функціональні підсистеми з вільними виходами – $Z_{9,11,12,16}$, $Z_{10,17}$, Z_2 і, відповідно, перевірками $П_{9,11,12,16}$, $П_{10,17}$ та $П_2$.

1. Кореневою вершиною графа ставимо перевірку на вільному виході блоку, входи якого пов'язані з найбільшою кількістю виходів інших блоків. У даному прикладі це донизу відходять два ребра – результати перевірки - з вагою (або вагами): логічна одиниця 1 - ліворуч (це означає, що усі блоки, пов'язані з цією перевіркою, працездатні), та логічний 0 - праворуч. Це свідчить, що якійсь з блоків, пов'язаних із цією перевіркою, непрацездатний (рис. 5а).

2. З вершини $П_{9,11,12,16}$ донизу ліворуч виходить ребро з вагою 1 (це означає, що попередні блоки Q_3, Q_1 - справні, стани S_3, S_1 - висячі вершини графа). Тоді треба перевірити другий вхід блока $Q_{9,11,12,16}$ - вихід блока Q_7, і вхід та вихід наступного блока Q_5. Поєднуємо це ребро з вершиною $П_5$, з якої праворуч виходить ребро з вагою 0. Це означає, що несправним є блок Q_5, тому останнє ребро (з вагою 0) закінчується висячою вершиною S_5.

3. Окремо розглядаємо фрагмент системи $Q_{9,11,12,16}$ (згідно рис. 3). Розриваємо зворотний зв'язок. За кореневу вершину приймаємо перевірку $П_{12}$ з вільного виходу Z_{12}. З вершини $П_{12}$ ліворуч виходить ребро з вагою 1 і далі по одиничних ребрах з вершин $П_{11}$, $П_{12}$, $П_{16}$ спускаємося до справного стану цієї підсистеми $S_{OГ}$ (рис. 5б).

4. Аналогічно слідкуємо за 1 від кожного вільного виходу до відповідного входу кожної підсистеми (рис. 5в, 5г). Потім повертаємося з висячих вершин до кореневої вершини, доробляючи нульові ребра.

5. Граф закінчено, коли число всіх висячих вершин дорівнює числу блоків функціональної схеми, або ж числу блоків прийнятої логічної моделі.

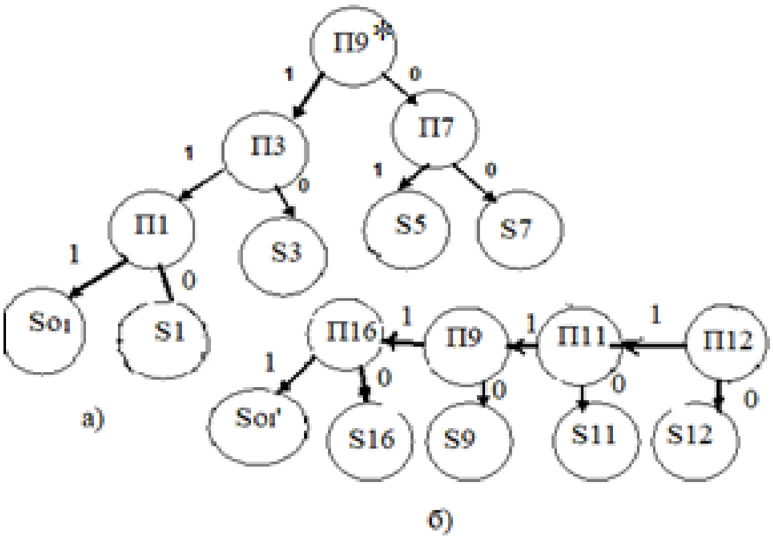
6. Якщо висячі вершини Si $\forall$ i = 1,…,*n* отриманого графа не повторюються, то цей граф побудований вірно.

7. Висота L тесту дорівнює максимальному числу ребер графа від кореневої вершини до висячої.

Оригінальність цього підходу заключається в тому, що виконується узагальнена логічна модель I типу, і з вільних виходів цієї моделі будується граф G2 (модель II типу).

Граф зручно та вигідно будувати, коли блоки системи значно відрізняються між собою по надійності (ймовірності відмов).

Використаний досвід закордонних дослідників з впровадження методів та алгоритмів глибокого навчання [11-16]. Програма розробки логічних моделей I та II типів бортової системи кондиціонування виконана на мові Python 3.12 у середовищі Visual Studio Code.



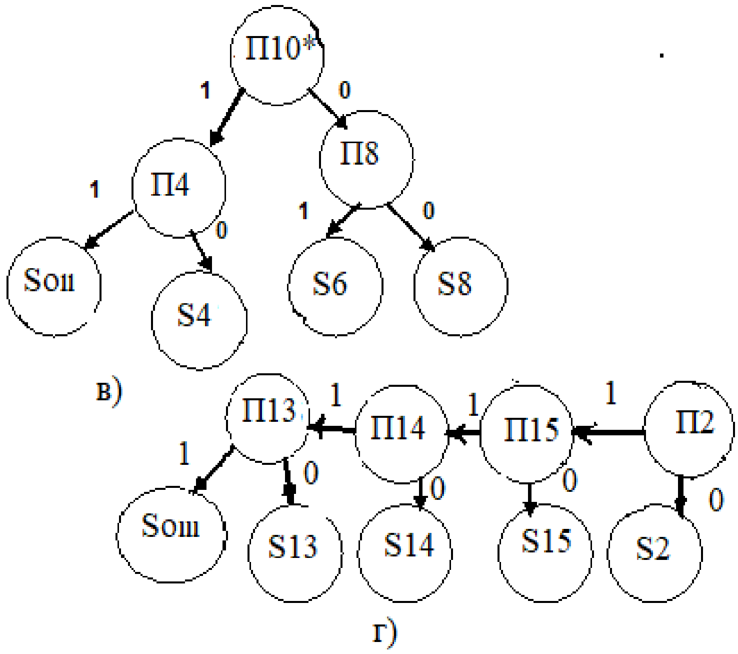


Рис. 5. Графи причинно-наслідкових зв'язків G2

Висновки

Дана робота дозволяє надати можливість оволодіти практичними знаннями та навичками побудови логічних моделей першого та другого типів, складати оптимальні тести працездатності та пошуку пошкоджень на етапах розробки, експлуатації та технічного обслуговування складних інформаційних систем. Метод глибокого навчання сприяє підвищенню точності моделі, швидкодії знаходження несправностей, передбачення та запобіганню аварій.

Література

1. Володарський Є. Т., Кухарчук В. В., Поджаренко В. О., Сердюк Г. Б. Метрологічне забезпечення вимірювань і контролю. Навчальний посібник. – Вінниця: Велес, 2001. – 219 с.

2. Пашко С. В., Сініцин І. П. Оптимальні рішення в системах, що складаються з раціональних агентів./ Artificial Intelligence, 2023, №2, с. 16-25.

3. Wooldridge M. An introduction to multiagent systems. John Wiley & Sons, 2009. – 348 p.

4. В. А. Голембо, О. Ю. Бочкарьов, А. М. Ціж. Задача формування індивідуальних зон відповідальності колективом мобільних агентів. — Львів: Львівська політехніка, 2006, с. 62-67. https://ena.lpnu.ua:8443/server/api/core/bitstreams/7997acc2-9a57-4660-a9ff-3eeb18cbf4c6/content

5. Pankratova N. System approach to estimation of guaranteed safe operation of complex engineering systems/ ITEC, 2010. https://mmsa.kpi.ua

6. Панкратова Н. Д. Гарантоване функціонування кіберфізичних систем в умовах концептуальної невизначеності/ Доповідь на XXIII Міжнародній науково-технічній конференції «Штучний інтелект та інтелектуальні системи» (AIIS’2023) 10 листопада 2023 р.

7. Системи кондиціювання - призначення і класифікація. 1.05.2019. Дата звернення 24.11.2024. https://karno.ua/ua/conditioners/sistemy-kondicionirovaniya/

8. Система кондиціювання повітря (авіація). 18.05.2022. Дата звернення 24.11.2024. https://uk.wikipedia.org/wiki/система_кондиціювання повітря(авіація).

9. Eitel, Elisabeth (6 May 2014). "CFD software models how moving parts affect aircraft-cabin airflow". Machine Design Magazine. Archived from the original on 1 July 2014.

10. Lehle W. Konzept und Entwicklung von Klimaanlagen für Flugzeuge / Praxis seminar Luftfahrt: Air Systems - Hamburg, 1. Juni 2006.- 6 S. http://humburg.dglr.de

11. Saman Taheri, Amirhossein Ahmadi, Behnam Mohammadi-Ivatloo, Somayeh Asadi. Fault detection diagnostic for HVAC systems via deep learning algorithms/ Energy and Buildings, Volume 250, 1 November 2021, 111275.

12. Russell S., Norvig, P. Artificial intelligence: a modern approach, 4th Edn. Hoboken, NJ: Pearson, 2021. - 1115 p.

13. Mathieu Cliché. 2017. BB twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs.

14. Raschka S., Mirjalili V. Python Machine Learning. Livery Place: Packt Publishing.-Ltd, 2019. 296 c.

15. Шаріфані, Куша та Аміні, Мах'яр. Машинне та глибоке навчання: огляд методів і застосувань (2023). World Information Technology and Engineering Journal, том 10, випуск 07, с. 3897-3904, 2023 р., доступний на SSRN: https://ssrn.com/abstract=4458723

16. Bagi, R., Dutta, T., Gupta, H.P. (2020). Deep Learning Architectures for Computer Vision Applications: A Study. In: Kolhe, M., Tiwari, S., Trivedi, M., Mishra, K. (eds) Advances in Data and Information Sciences. Lecture Notes in Networks and Systems, vol 94. Springer, Singapore. https://doi.org/10.1007/978-981-15-0694-9_56

References

1. Volodarskyi E. T., Kuharchuk V. V., Podzharenko V. O., Serdyuk G. B. (2001). Metrological support of measurements and control. Study guide. – Vinnytsia: Veles.

2. Pashko S. V., Sinitsyn I. P. (2023). Optimal solutions in systems consisting of rational agents / Artificial Intelligence, No. 2.

3. Wooldridge M. (2009). An introduction to multiagent systems. John Wiley & Sons.

4. V. A. Golembo, O. Yu. Bochkarev, A. M. Cizh. (2006). The task of forming individual zones of responsibility by a team of mobile agents. - Lviv: Lviv Polytechnic. https://ena.lpnu.ua:8443/server/api/core/bitstreams/7997acc2-9a57-4660-a9ff-3eeb18cbf4c6/content

5. Pankratova N. (2010). System approach to estimation of guaranteed safe operation of complex engineering systems/ ITEC. https://mmsa.kpi.ua

6. Pankratova N. D. (2023). Guaranteed functioning of cyber-physical systems in conditions of conceptual uncertainty/ Report on the XXIII International Scientific and Technical Conference «Artificial Intelligence and Intellectual Systems» (AIIS'2023).

7. Air conditioning systems - purpose and classification, (2019). https://karno.ua/ua/conditioners/sistemy-kondicionirovaniya/

8. Air conditioning system (aviation). (2022). https://uk.wikipedia.org/wiki/air_conditioning_system_(aviation)

9. Eitel, Elisabeth (2014). CFD software models how moving parts affect aircraft-cabin airflow/ Machine Design Magazine.

10. Lehle W. (2006). Concept and development of air conditioning systems for aircraft / Practical seminar on aviation: Air Systems – Hamburg. http://humburg.dglr.de

11. Saman Taheri, Amirhossein Ahmadi, Behnam Mohammadi-Ivatloo, Somayeh Asadi. (2021). Fault detection diagnosis for HVAC systems via deep learning algorithms/ Energy and Buildings, Volume 250.

12. Russell S., Norvig, P. (2021). Artificial intelligence: a modern approach, 4th Edn. Hoboken, NJ: Pearson.

13. Mathieu Cliché. (2017). BB twtr at SemEval-2017 Task 4: Twitter Sentiment Analysis with CNNs and LSTMs.

14. Raschka S., Mirjalili V .(2019). Python Machine Learning. Livery Place: Packt Publishing.-Ltd.

15. Sharifani, Kusha and Amini, Mahyar. (2023). Machine and Deep Learning: A Review of Methods and Applications. World Information Technology and Engineering Journal, Volume 10, Issue 07. https://ssrn.com/abstract=4458723

16. Bagi, R., Dutta, T., Gupta, H.P. (2020). Deep Learning Architectures for Computer Vision Applications: A Study. In: Kolhe, M., Tiwari, S., Trivedi, M., Mishra, K. (eds) Advances in Data and Information Sciences. Lecture Notes in Networks and Systems, vol 94. Springer, Singapore. https://doi.org/10.1007/978-981-15-0694-9_56

The article has been sent to the editors 18.10.24.
After processing 20.11.24.
Submitted for printing 30.12.24.